### Solving analytical queries on Redshift Cluster

Here, you have to write the query used for solving the question and the screenshots of the table which is outputted after the query is run on the AWS Redshift Query editor UI.

1. **Top 10 ATMs where most transactions are in the ’inactive’ state**

select f.atm\_id, a.atm\_number, count(f.trans\_id) as total\_trans\_count, l.location

from spar\_nord\_bank\_atm\_data.fact\_atm\_trans f

inner join spar\_nord\_bank\_atm\_data.dim\_atm a on a.atm\_id = f.atm\_id

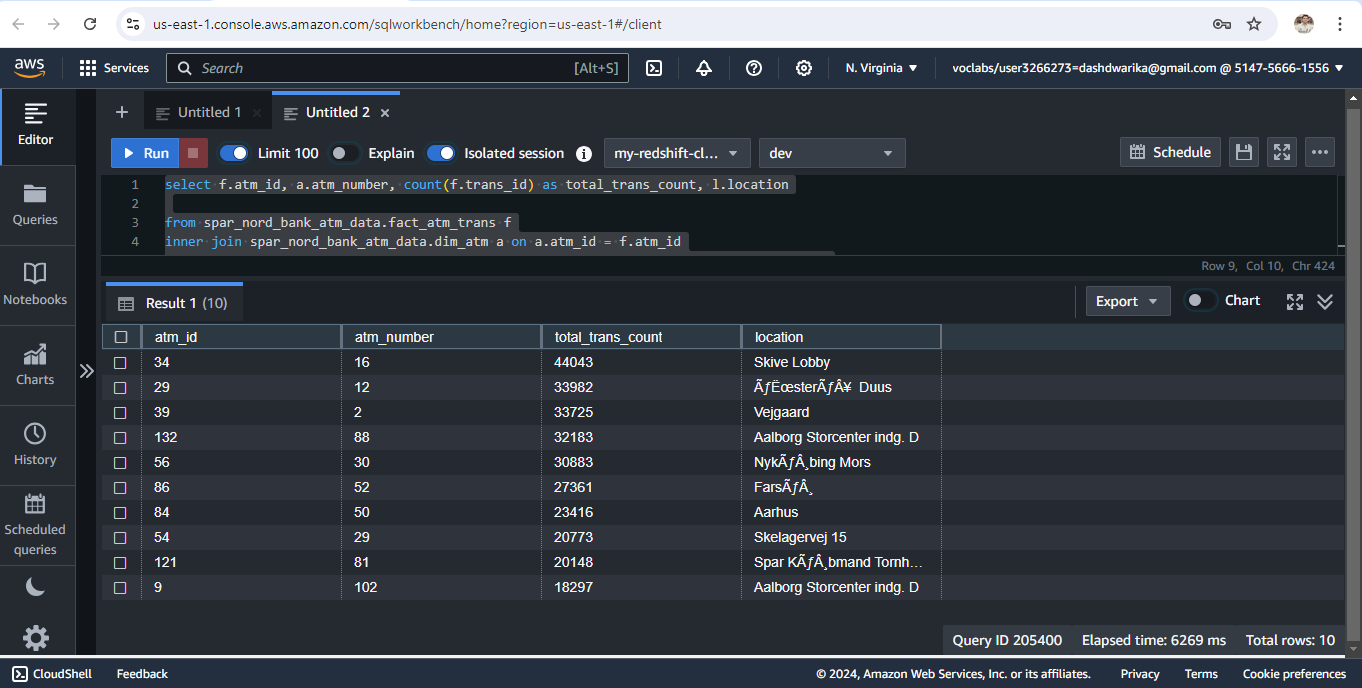
inner join spar\_nord\_bank\_atm\_data.dim\_location l on a.atm\_location\_id = l.location\_id

group by f.atm\_id, a.atm\_number, l.location, f.atm\_status

having f.atm\_status = 'Inactive'

order by total\_trans\_count DESC

Limit 10;



1. **Number of ATM failures corresponding to the different weather conditions recorded at the time of the transactions**

WITH weather\_atm\_failure AS (

select weather\_main, count(trans\_id) as total\_trans\_count, sum(case when atm\_status = 'Inactive'then 1 else 0 end) as atm\_status\_inactive\_count

from spar\_nord\_bank\_atm\_data.fact\_atm\_trans

where weather\_main != ''

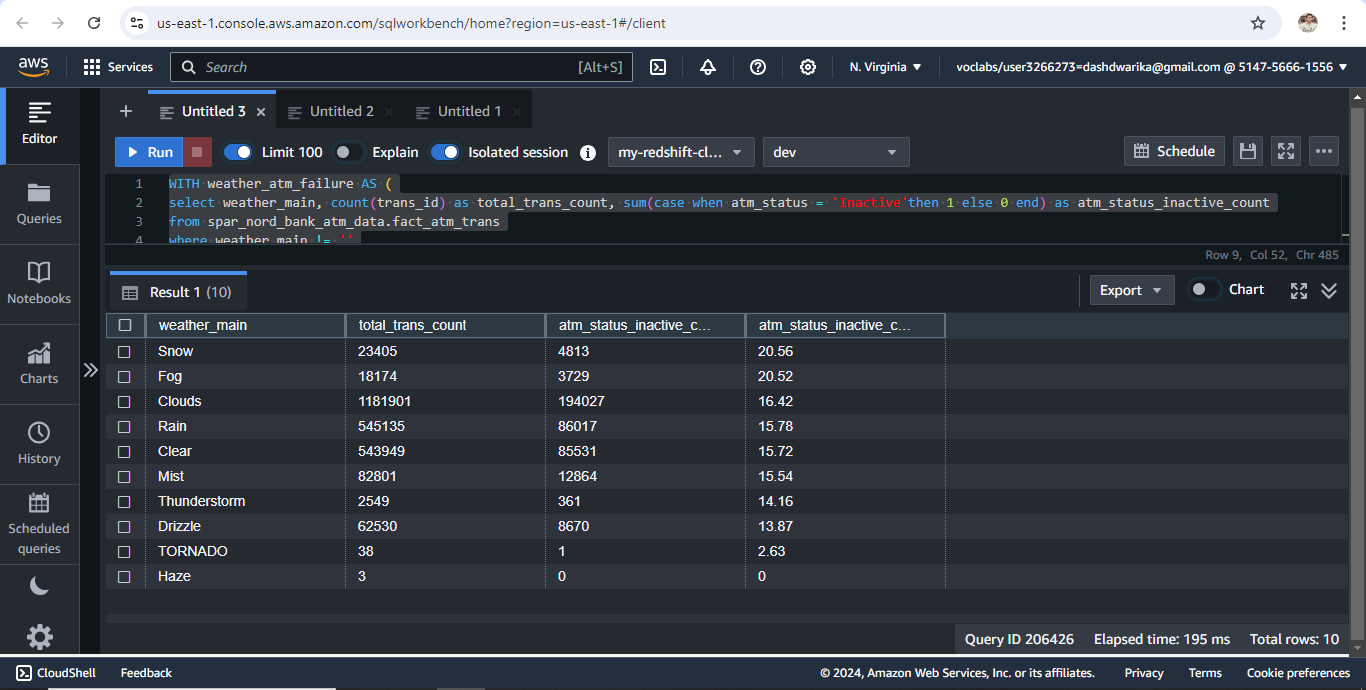
group by weather\_main

)

select \*, round(((CAST(atm\_status\_inactive\_count as numeric) / total\_trans\_count) \* 100), 2) AS atm\_status\_inactive\_count\_percentage

from weather\_atm\_failure

order by atm\_status\_inactive\_count\_percentage DESC;



1. **Top 10 ATMs with the most number of transactions throughout the year**

select f.atm\_id, a.atm\_number, a.atm\_manufacturer, l.location, count(f.trans\_id) as total\_trans\_count

from spar\_nord\_bank\_atm\_data.fact\_atm\_trans f

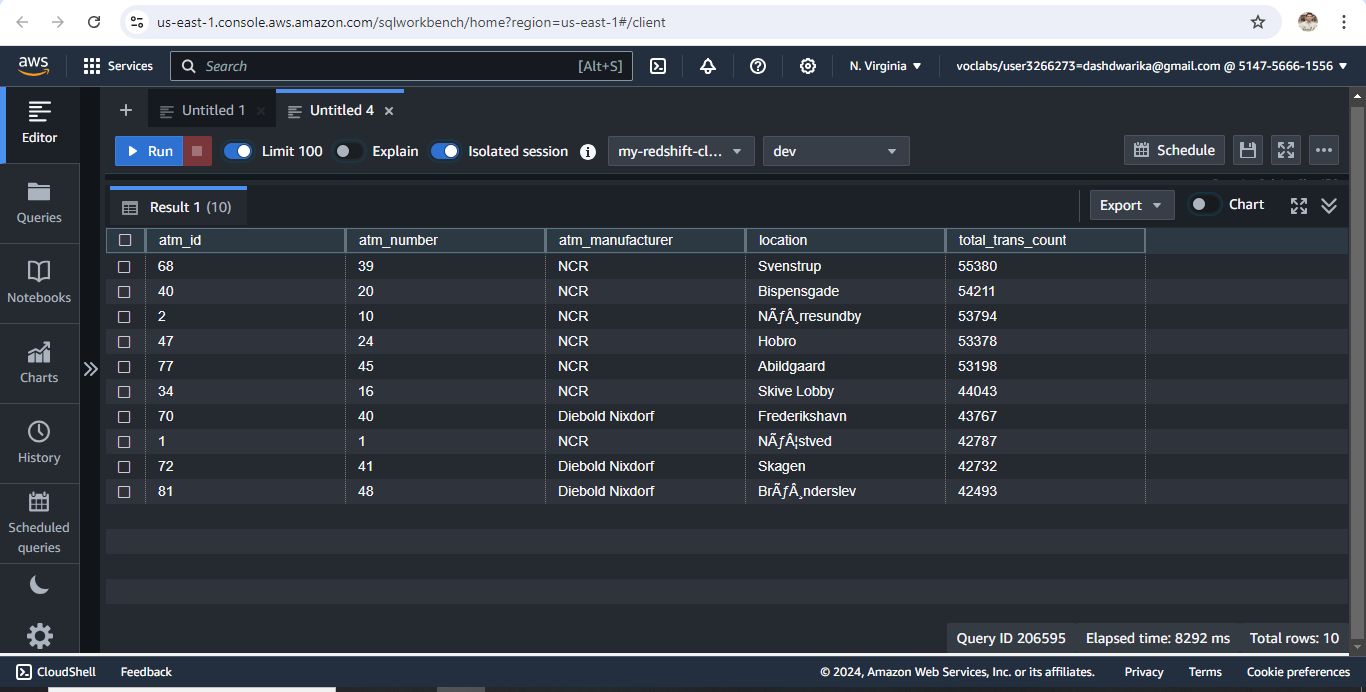
inner join spar\_nord\_bank\_atm\_data.dim\_atm a on a.atm\_id = f.atm\_id

inner join spar\_nord\_bank\_atm\_data.dim\_location l on a.atm\_location\_id = l.location\_id

group by f.atm\_id, a.atm\_number, a.atm\_manufacturer, l.location, f.atm\_status

order by total\_trans\_count DESC

Limit 10;

****

1. **Number of overall ATM transactions going inactive per month for each month**

WITH inactive\_count\_month\_wise AS (

select d.month, count(f.trans\_id) as total\_trans\_count, sum(case when f.atm\_status = 'Inactive'then 1 else 0 end) as atm\_status\_inactive\_count

from spar\_nord\_bank\_atm\_data.fact\_atm\_trans f

inner join spar\_nord\_bank\_atm\_data.dim\_date d on d.date\_id = f.date\_id

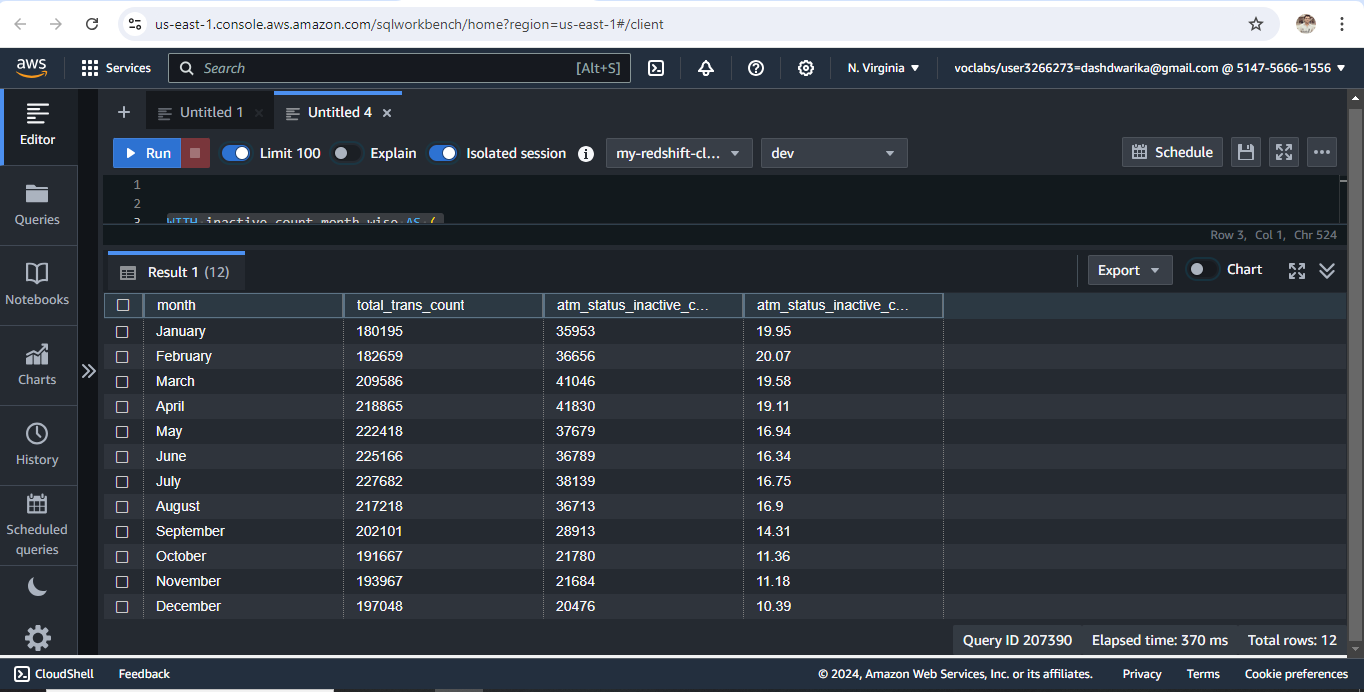
group by d.month

)

select \*, round(cast(atm\_status\_inactive\_count as NUMERIC) / total\_trans\_count \* 100, 2)  as atm\_status\_inactive\_count\_percent

from inactive\_count\_month\_wise

order by to\_date(month, 'Month');

****

1. **Top 10 ATMs with the highest total withdrawn amount throughout the year**

select f.atm\_id, a.atm\_number, a.atm\_manufacturer, l.location, sum(f.transaction\_amount) as total\_trans\_amount

from spar\_nord\_bank\_atm\_data.fact\_atm\_trans f

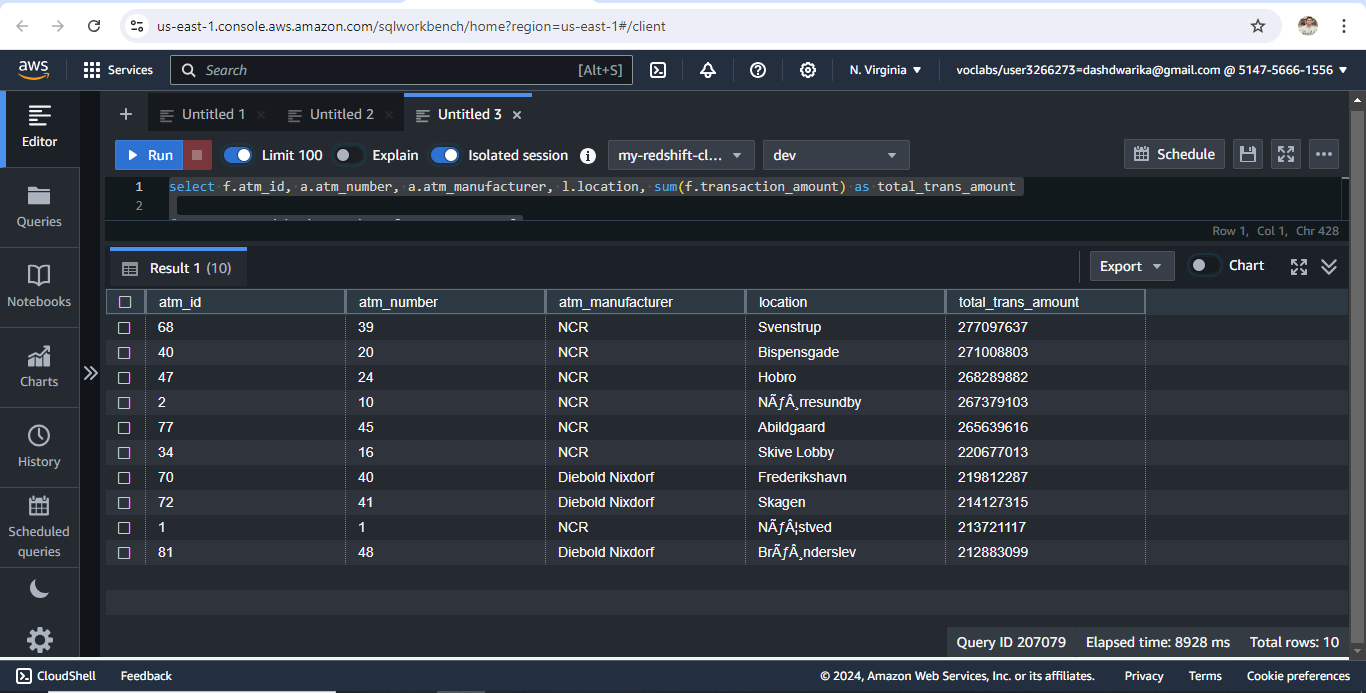
inner join spar\_nord\_bank\_atm\_data.dim\_atm a on a.atm\_id = f.atm\_id

inner join spar\_nord\_bank\_atm\_data.dim\_location l on a.atm\_location\_id = l.location\_id

group by f.atm\_id, a.atm\_number, a.atm\_manufacturer, l.location

order by total\_trans\_amount DESC

Limit 10;



1. **Number of failed ATM transactions across various card types**

WITH inactive\_count\_card\_wise AS (

select c.card\_type, count(f.trans\_id) as total\_trans\_count, sum(case when f.atm\_status = 'Inactive'then 1 else 0 end) as atm\_status\_inactive\_count

from spar\_nord\_bank\_atm\_data.fact\_atm\_trans f

inner join spar\_nord\_bank\_atm\_data.dim\_card\_type c on c.card\_type\_id = f.card\_type\_id

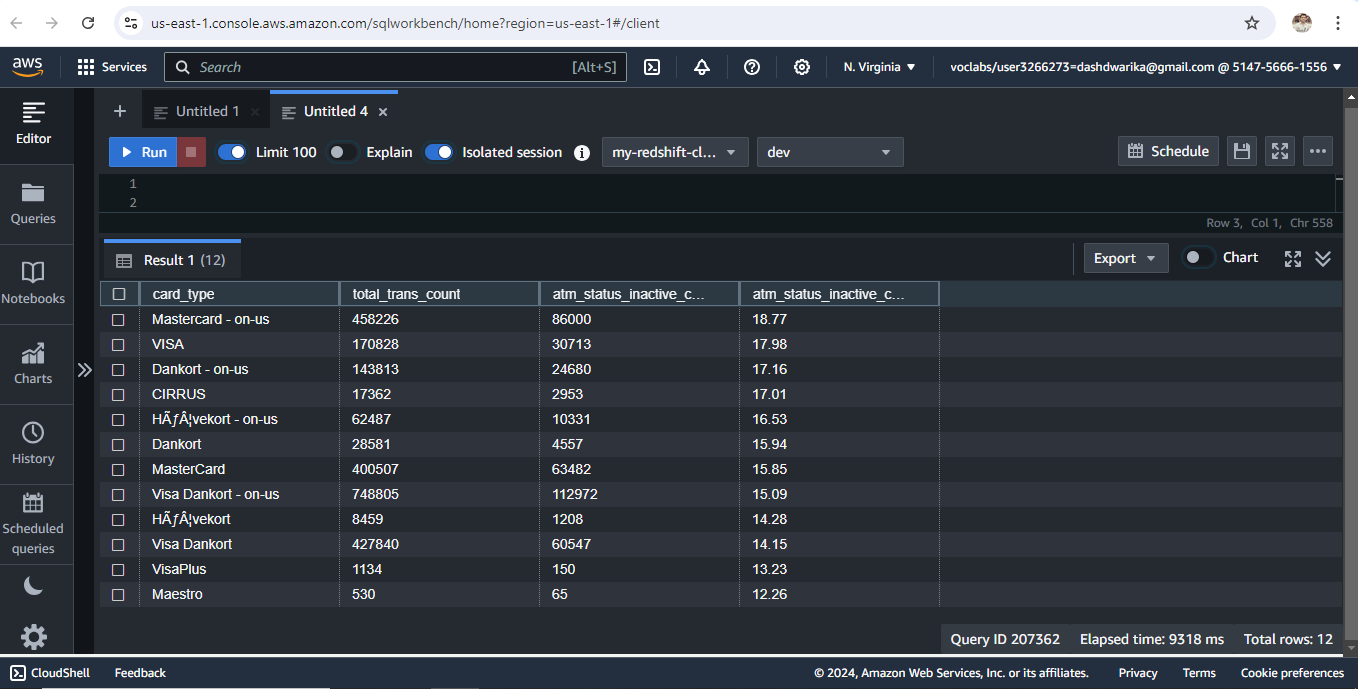
group by c.card\_type

)

select \*, round(cast(atm\_status\_inactive\_count as NUMERIC) / total\_trans\_count \* 100, 2) as atm\_status\_inactive\_count\_percent

from inactive\_count\_card\_wise

order by atm\_status\_inactive\_count\_percent DESC;



1. **Number of transactions happening on an ATM on weekdays and on weekends throughout the year. Order this by the ATM\_number, ATM\_manufacturer, location, weekend\_flag and then total\_transaction\_count**

select a.atm\_number, a.atm\_manufacturer, l.location, (case when d.weekday in ('Saturday', 'Sunday') then 1 else 0 end) as weekend\_flag, count(f.trans\_id) as total\_trans\_id

from spar\_nord\_bank\_atm\_data.fact\_atm\_trans f

inner join spar\_nord\_bank\_atm\_data.dim\_atm a on a.atm\_id = f.atm\_id

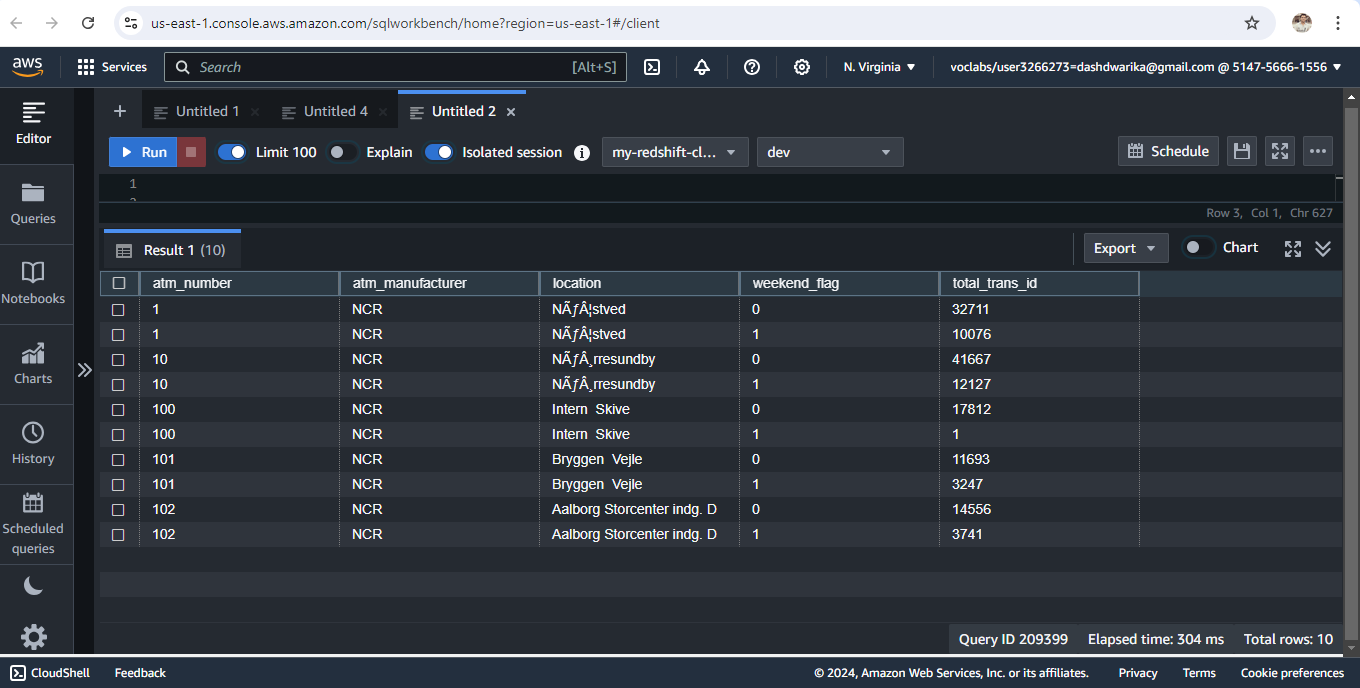
inner join spar\_nord\_bank\_atm\_data.dim\_location l on l.location\_id = a.atm\_location\_id

inner join spar\_nord\_bank\_atm\_data.dim\_date d on d.date\_id = f.date\_id

group by a.atm\_number, a.atm\_manufacturer, l.location, weekend\_flag

order by a.atm\_number, a.atm\_manufacturer, l.location, weekend\_flag, total\_trans\_id DESC

LIMIT 10;



1. **Most active day in each ATMs from location "Vejgaard"**

WITH vejgaard\_atm\_trans\_data AS (

select a.atm\_id, a.atm\_number, a.atm\_manufacturer, l.location, count(f.trans\_id) as total\_trans\_id, d.weekday

from spar\_nord\_bank\_atm\_data.fact\_atm\_trans f

inner join spar\_nord\_bank\_atm\_data.dim\_atm a on a.atm\_id = f.atm\_id

inner join spar\_nord\_bank\_atm\_data.dim\_location l on l.location\_id = a.atm\_location\_id

inner join spar\_nord\_bank\_atm\_data.dim\_date d on d.date\_id = f.date\_id

where l.location = 'Vejgaard'

group by a.atm\_id, a.atm\_number, a.atm\_manufacturer, l.location, d.weekday

order by total\_trans\_id DESC

)

select \* from vejgaard\_atm\_trans\_data v1

where v1.total\_trans\_id = (select max(total\_trans\_id) from vejgaard\_atm\_trans\_data v2 where v1.atm\_id = v2.atm\_id);

